PCT patent application
in the name of
NOKIA Corporation

TITLE OF THE INVENTION

Multi-Stream FFT for MIMO-OFDM systems

FIELD OF THE INVENTION

The present invention relates to a processor and method for
subjecting multiple parallel input data streams to Fast
Fourier Transformation, FFT.

BACKGROUND OF THE INVENTION

By using Fast Fourier Transformation, the Discrete Fourier
Transform can be obtained. This is important in many signal
processing scenarios.

In particular in, for example, mobile communication
scenarios, the FFT is required to be obtained for various
purposes. Conventionally, in case a single data stream is
to be subjected to FFT transformation, various scenarios
for accomplishing this are known. A single data stream is
often referred to as SISO, "Single Input Single Output". As
a typical SISO scenario, one might consider a case in which
a communication network entity such as a base station or
Node_B transmits via a single antenna or antenna element
data to a mobile station or user equipment with one antenna
element(or vice versa).

2

On the other hand, with further developments in
communication technology, scenarios are implemented and
under investigation which apply multiple antenna elements
for transmission and for reception. In such cases, a so-
5   called "Multiple Input Multiple Output", MIMO, concept is
present. MIMO concepts are often applied in connection with
Orthogonal Frequency Division Multiplex, OFDM, systems.

MIMO-OFDM (multiple-input-multiple-output orthogonal
10   frequency division multiplex) systems offer remarkable
increase in link reliability and/or in data rate. However,
this new technique suffers on higher complexity of the
hardware. For this reason, there is a need of clever
strategies to reduce the expenditure of hardware.
15
Apparently, with multiple input data streams being present
simultaneously, i.e. in parallel, also those multiple data
streams have to be subjected to FFT. This imposes a certain
problem in terms of processing load, processing speed,
20   and/or complexity for the signal processing methods and
hardware used for this purpose.

The FFT transformation is a central process in conventional
OFDM (SISO-OFDM: single-input-single-output OFDM) systems.
25   The transition to MIMO technique results in an OFDM system
with several FFT transformation processes in parallel. For
instance, MIMO systems with four receiver antenna elements
need four FFT transformations. In straightforward
solutions, there have to be installed four FFT processing
30   blocks. This leads to much higher hardware complexity.
Hence, there is a need for a new implementation strategy of
the FFT for MIMO systems.

He and Torkelson have presented "A new approach to Pipeline
35   FFT processor" in IEEE Proceeedings of IPPS '96, 1996, pp.

3

766 to 770. This document introduces various pipeline FFT
processors for SISO scenarios.

For better understanding of the present invention to be
5   described hereinafter, a brief review and introduction of
the FFT pipeline architecture as presented by He and
Torkelson is given hereinafter. A particular usable FFT is
briefly introduced to obtain an idea of the main structure
and its properties.
10

To this end, the SISO Radix 2^2 single-path delay feedback
(SDF) architecture proposed by He & Torkelson will be
considered. This architecture is also referred to as
R2^{2}SDF.
15

FFT for SISO Systems according to He & Torkelson

As mentioned, a structure of the FFT algorithm was
proposed, where a Radix 2^2 single-path delay feedback (SDF)
20   architecture is used. Because of the SDF, the spatial
regularity of the resulting architecture / signal flow
graph could be exploited. The resulting hardware
requirement is minimal on both dominant components: complex
multipliers and complex data memory.
25

For a hardware-oriented implementation, this approach
combines the advantage of the signal flow graph, SFG, of
radix 4 and radix 2 approaches. The SFG radix 4 requires
minimum of non-trivial multipliers, whereas the SFG radix 2
30   uses a simple butterfly structure.

Figure 1 illustrates the resulting signal flow graph
structure for N=16 (16 points FFT), i.e. a received data
stream to be subjected to FFT is assumed to comprise N=16
35   samples (N samples forming one symbol). Trivial

4

multiplications denoted by the multiplier "-j" appear
between a first, BF **I**, and a second, BF **II**, stage of the
SFG. At the first stage, a simple butterfly structure is
used. Then, in the second stage, the same calculation
5   process is realized. And additionally, the last $N/4=4$
outputs of the first stage BF**I** are multiplied by -j.
Assuming a complex number $Z = R + j*I$ with R denoting the
real component and I denoting the imaginary component, a
multiplication by "-j" will then lead to $-j*Z = -j*R + I$.
10   Apparently, the real and imaginary parts are exchanged and
the imaginary part is inverted in terms of the sign.
Therefore, this multiplication is regarded as trivial
(real-imaginary swapping and sign inversion). These
operations are indicated by diamonds symbols in Figure 1.
15   After these two stages, full multipliers are required to
compute the product of the decomposed twiddle factor. The
multipliers perform a multiplication with multiplication
factors W (twiddle factors). Twiddle factors are those
coefficients applied to results from a previous stage to
20   combine these in order to form inputs of a next stage.

Applying the Common Factor Algorithm, CFA, procedure
recursively to the remaining DFT's (Discrete Fourier
Transforms) of lengths $N/4$, the complete radix 2^2 DIF FFT
25   algorithm is obtained, as shown in Figure 2. As an
explanatory remark, using such an approach, a number of
$N=16$ data sets (samples) of an incoming stream is
decomposed in a pipeline fashion into a succession of
stages $\log_2 N = 4$. That is, for $N=16$ data samples, a 4 stage
30   FFT SFG and/or architecture will result (totoal number of
stages $k=4$ in this example). A respective i-th stage
$(i=1...4)$ is designed to process a number of data sets of
$2^{(\log_2 N+1-i)}$. Thus, the first stage $(i=1)$ BF **I**
receives/processes 16 data samples, and the fourth stage
35   $(i=4)$ BF **IV** receives/processes 2 data samples.

Architecture

In the following, the architecture will be described with
5   reference to a DFT example for N=16 samples.

As shown in Fig. 2, the FFT structure for N=16 data samples
has four butterfly stages BFI, …, BFIV. Note that BFI, …,
BF IV denote the stages and do not denote the BF types
10   employed in a respective stage. There can be seen that the
non-trivial multipliers are between the second, BFII, and
the third stage, BFIII, according to the signal processing
order. In addition, the rotations (trivial multiplications)
by −j are done after the first, BFI, and after the third,
15   BFIII, stage. Fig. 3 illustrates the resulting pipeline
architecture. The blocks above the butterfly structures
indicate FIFO memories and the numbers indicated therein
the delay imposed thereby, i.e. number of samples buffered
by these.
20
The FIFO memories are located in the single delay feedback
path of the structure. FIFO memories are particularly
useful in terms of hardware, but the FIFO property could
also be realized by another memory type in combination with
25   appropriate addressing of the memory in order to read out
the stored data in FIFO fashion.

For instance, the FIFO in the first stage after the input
port has the length of 8 symbols. Apparently, the number of
30   delay elements, i.e. the number of samples buffered in the
feedback path of a i-th stage out of k stages is N/2 for
i=1, N/4 for i=2, N/8 for i=3, and N/16 for i=4, and can
generally be expressed as $N/2^i$ for an i-th stage.
The data control for the butterflies is indicated by the
35   bar on the bottom of the figure, which schematically

indicates control signals supplied to the four stages 1...4
of the pipeline architecture. Butterfly stages of type I
(BF2I) receive a single control signal only and are applied
in stages i=1 and i=3, and Butterfly stages of type II
5   (BF2II) receive two control signals and are applied in
stages i=2 and i=4. The twiddle factors W(n) are for
example read out from a memory (not shown in Fig. 3) with
appropriate timing. The timing of the control signals
supplied to BF2I and BF2II stages as well as for twiddle
10   factor generation/supply depends on the clock rate of the
FFT device.

The internal structure of the respective butterfly stage is
shown in Fig. 4 (BF2I) and Fig. 5 (BF2II). Note that input
15   and output ports are divided into a real (index r) and
imaginary (index i) part. N denotes the number of symbols
contained in the stream to be subjected to FFT processing
and n is an index variable with $1<=n<=N$. (The memory
"capacity" of e.g. the FIFO in the feedback path depends on
20   the stage index i with $1<=i<=k$.)

Fig. 11A and 12 show details of the data control in terms
of control signals applied and timing relations there
between, as will be described later on.
25

The calculation process at each stage is done in two steps.

In the first step (control signal s = 0), the data sequence
x(n) (n=1..16/2) is read at the input ports
30     $x_r(n+N/2)$/$x_i(n+N/2)$ and is directly written to the ports
$Z_r(n+N/2)$/$Z_i(n+N/2)$ which are connected to the FIFO. At the
same time, the FIFO content is read at the ports
$x_r(n)$/$x_i(n)$ and is directly written, as the other output
port pair, to the ports $Z_r(n)$/$Z_i(n)$ which are connected to
35   the next pipeline stage.

7

In the second step(control signal s = 1), after N/2=8
symbols, , the stored data and the remaining input symbols
$x(n)$ (n=9..16) are used to compute the stage output where

5   one half is written to the next stage (ports $Z_r(n)/Z_i(n)$)
and the other half is stored in the FIFO memory (ports
$Z_r(n+N/2)/Z_i(n+N/2)$).

To accomplish such processing, the internal structure uses

10 adders/subtractors and internal signal feeding paths as
shown in Fig. 4. In addition, supplying the signals to FIFO
memory and/or next stage Butterfly stage is accomplished
using switches under control of the control signal s. The
operational condition of a respective switch is denoted by

15 0 and/or 1 which represents the respective state of the
control signal s applied in order for the switch to be in
the respective operational condition. An adder is
illustrated by the encircled "+", a subtractor is
illustrated by the encircled "+" with an additional

20 subscript "-".

The calculation process of the butterfly stage BF2II
differs from the one done in BF2I a little. Since these
stages additionally include the j rotations, i.e. the

25 "trivial" multiplications by "-j", the real and imaginary
parts of input signals have to be swapped. In addition, the
signs have also to be changed as shown in Fig. 5. This is
controlled by the signal t. The negated signal t is
logically combined in an AND gate with the signal s and

30 controls the swapping paths at the input terminals
xr(n+N/2), xi(n+N/2) as well as the adders/subtractors in
the signal paths associated to the signals xi(n) and
xi(n+N/2). Thus, for s=1 and t=0 there occurs a swapping
and conversion of the adder, else there is no swapping and

8

conversion of the adder. The remaining process and
architecture is equal to the BFI process.

Fig. 11A shows details of control signals with a
5   corresponding timing relation being illustrated in Fig. 12.

As shown in Fig. 11A, a clock signal clk is supplied to the
(FIFO) memory, a twiddle factor generation means (e.g.
including a memory from which the factors are read out) and
10   the BF2II stage. A signal supplied to the BF2II stage from
a preceding stage is denoted with x, and signals s and t as
explained before are also supplied. A signal leaving the
BF2II stage to a subsequent multiplier is denoted with z
and supplied to the multiplier for multiplication with a
15   twiddle factor w. Afterwards, the multiplied signal is
forwarded to the next stage (not shown in Fig. 11A). (Note
that substantially the same holds for a stage of type BF2
I, with the difference that the control signal t is not
applied and that a signal z leaving a stage of BF2I type
20   will be supplied to a BF2II stage (input signal x) and not
to multiplier performing multiplication with twiddle
factors).

Fig. 12 shows the timing relation there between.
25   In the lower part of Fig. 12, the signals z, w and clk are
supplied in synchronism with each other. With each clock
cycle clk, a new signal z is supplied to the multiplier
which is in synchronism therewith supplied with a
corresponding weight (twiddle) factor w.
30   In the upper part of Fig. 12 it is shown that a sample x of
a sequence of 1 ... N samples (forming one OFDM symbol) is
supplied with each clock cycle clk. Initially, the signal s
assumes a low level (s=0) for the first N/2 samples.
Thereafter, starting with sample N/2+1, it assumes a high
35   level until N samples have been supplied. (Thereafter, a

new OFDM symbol sequence starts and s=0). As to the signal t, this signal assumes a high level for the first $3*N/4$ samples and changes afterwards (starting with sample $3/4*N+1$) for the last $N/4$ samples to the low level.

Finally, Table 1 shows the complexity of this prior art FFT architecture, which is used in the further development of the multi-stream transformation for MIMO-OFDM systems.

	Multiplier	Adder	Memory Size	Control
R2²SDF	$Log_4 \ N_{FFT} \ -1$	$4Log_4 \ N_{FFT}$	$N_{FFT} \ -1$	Simple

Table 1: Computational Complexity of the FFT.

FFT for MIMO Systems

Now, two straightforward architecture alternatives are presented for MIMO systems based on this FFT structure. Notwithstanding this, other FFT structures could be used. In the following, the previously described FFT structure (R2²SDF) is implemented for MIMO systems. There are two possible strategies to realize the transformation process for M_R antenna system, i.e. systems having a number of M_R antennas.

Fig. 6 shows a full parallel implementation with a FFT block per each data stream to be transformed. Thus, on the one hand, a number M_R of FFT blocks can be implemented, i.e. one for each stream (see Fig. 6 for the example of $M_R=4$). It can be seen that the complexity of such a system grows linearly with the number of antennas (i.e. M_R times one FFT complexity).

On the other hand, to reduce the complexity of the system, the transformation process can be done successively by a smaller number (M_{FFT}) of FFT blocks (straightforward successive FFT solution). In order to transform

5 successively M_R parallel streams, the FFT has (or the FFTs have) to work at a higher rate. Because of the used FFT pipeline structure, the frequency can be increased arbitrarily.

10 Fig. 7 illustrates such a successive transformation process for $M_R=4$ and $M_{FFT}=1$, i.e. using a single FFT only. Due to this processing, the input streams are multiplexed upstream of the FFT using a multiplexer MUX and demultiplexed using a demultiplexer DeMUX after, i.e. downstream the FFT. This

15 strategy results in a reduction of computational complexity, depending on the sharing ratio (M_R/M_{FFT}). Unfortunately, each stream requires an additional input buffer that collects one OFDM symbol before sending it to the FFT.

20
Fig. 8 illustrates the timing of signal processing of this structure as shown in Fig. 7. In a first step, N_{FFT} symbols of each stream (example: number of streams $M_R=4$) are written to the corresponding stream buffer. Due to the M_R

25 streams arriving in parallel, the M_R buffers are simultaneously getting filled. Finally, after the buffering period, each buffer successively shifts its content into the FFT block, which works at a higher rate. Since the buffer content of the streams is used sequentially and new

30 data symbols are continuously fed to the FFT at the same time, another buffer (not shown) is needed.

In a first buffer area I, samples of M_R data streams are buffered. Assuming a multiplexing sequence of M_R streams

35 1...4, the samples of stream 1 are used as FFT input first.

11

In the meantime, further data samples of following symbols
are buffered in a buffer area II for streams 2…4. Samples
of stream 2 will be subjected to FFT processing next, which
is the reason why buffer area II for stream 2 will not fill
5   too much. Since streams 3 and 4, respectively, will be
subjected to FFT processing pre-last or last, respectively,
the respective buffer area II for these streams will be
filled to a greater extent. The indication of multiples of
N_{FFT} indicate the additional amount of buffer memory
10   required for buffer area II.

The need and the size for the additional buffer area can
also be seen at the time axis t in Fig. 8. At the time when
the first sequence is fed into the FFT, the incoming values
15   of the remaining sequences have to be buffered until the
FFT block has finalized the input process for the first
sequence. For the second sequence for $M_R=4$, the FFT is able
to read the next sequence after $N/M_R=0.25N$ time steps. This
results in an absolute value of $t=1.25N$. For the 3rd and 4th
20   sequences, the waiting or buffer time is $2N/M_R=0.5N$
(absolute: $t=1.5N$) and $3N/M_R=0.75N$ (absolute: $t=1.75N$).
Consequently, the data input for all sequence is finalized
after N time steps and at the time $t=2N$ the next OFDM
symbol period begins.
25

Assuming an FFT processing rate of four times higher
compared to the symbol rate, the additional memory size for
buffering is

30
$$\frac{1}{2}\left(\frac{M_R^2}{M_{FFT}} - M_R\right)\frac{N_{FFT}}{4} \qquad\qquad \text{Eq. (1)}$$

12

In addition, the FFT uses a memory in the size of $N_{FFT}-1$. Thus, the overall memory size (complex symbols) is given by

$$\underbrace{M_R N_{FFT}}_{BufferI} + \underbrace{\left(\frac{M_R^2}{M_{FFT}} - M_R\right)\frac{N_{FFT}}{8}}_{BufferII} + \underbrace{(N_{FFT}-1)M_{FFT}}_{FFT} \qquad \text{Eq. (2)}$$

For a system with four antennas ($M_R=4$) and one FFT ($M_{FFT}=1$), the above equation can be simplified to

$$\underbrace{4N_{FFT}}_{BufferI} + \underbrace{1.5N_{FFT}}_{BufferII} + \underbrace{(N_{FFT}-1)}_{FFT} = 6.5N_{FFT}-1 \qquad \text{Eq. (3)}$$

For MIMO receivers with M_R antennas, M_R independent data symbol streams have to be transformed. Usually, according to the approach introduced with reference to Fig. 6, the data symbols are fed into M_R FFT blocks. Especially for large FFT length, this results in highly complex system architectures.

As shown in the successive processing alternative introduced with reference to Figs. 7 and 8, there is a possibility to reduce the architecture complexity up to a complexity of one FFT. Unfortunately, the memory consumption of this option increases from $4N_{FFT}-4$ (parallel FFTs solution) to $6.5N_{FFT}-1$ complex symbols.

SUMMARY OF THE INVENTION

Hence, it is an object of the present invention to provide an improved signal processor for FFT transformation as well as a corresponding method which is free from above mentioned drawbacks inherent to known approaches.

According to the present invention, this object is for
example achieved by
a signal processor for Fast Fourier Transformation, FFT, of
5   M_R, $M_R > 1$, input data streams supplied in parallel,
comprising a multiplexing device having M_R input terminals
each receiving one of the M_R input data streams and an
output terminal at which the M_R input data streams are
output in a multiplexed manner, a Fast Fourier
10   Transformation device configured to perform Fast Fourier
Transformation of a data stream supplied at an input
terminal thereof and to output the FFT transformed data
stream at an output terminal thereof, the input terminal of
the Fast Fourier Transformation device being connected to
15   the output terminal of the multiplexing device, and a
demultiplexing device having an input terminal connected to
the output terminal of the Fast Fourier Transformation
device and M_R output terminals at which a respective one of
M_R transformed output data streams is output in a
20   demultiplexed manner, characterized in that each of the M_R
input data streams contains a number of $N=2^k$ samples, the
Fast Fourier Transformation device has a pipeline
architecture composed of k stages with a respective
feedback path including a single delay element per each
25   stage of the pipeline architecture and is controlled by a
first and second internal control signals, wherein the
delay element in a feedback path of an ith stage, $1<=i<=k$,
of the pipeline architecture imposes a delay of $M_R*N/2^i$
samples, the first internal control signal is clocked M_R
30   times faster compared to a clock rate at which the samples
of the M_R streams are supplied, and the second internal
control signals are clocked M_R times slower compared to the
first internal control signal.

14

According to advantageous further developments of the
signal processor,

- the multiplexing device is configured such that the
M_R input data streams are multiplexed per data sample of
5   the input data streams and the demultiplexing device
(DEMUX) is configured such that the transformed input data
stream is demultiplexed per data sample of the transformed
data stream;

- a control signal supplied to the multiplexer and
10   demultiplexer is clocked at a rate M_R times the clock rate
of the supplied streams;

- the Fast Fourier Transformation device (FFT) has a
Radix-2 Single-path Delay Feedback, R^2SDF, architecture;

- the pipeline architecture of the Fast Fourier
15   Transformation device is composed of Butterfly stages of
types I and II;

- the first stage of the pipeline architecture
receiving the multiplexed data streams is a Butterfly stage
of type I for even and odd total numbers of k.

20

According to the present invention, further a network
element of a communication network comprising a signal
processor according to any of the preceding aspects is
concerned.

25

According to the present invention, further a terminal
configured to communicate via a communication network, the
terminal comprising a signal processor according to any of
the preceding aspects is concerned.

30

Still further, according to the present invention, a system
comprising at least one of a terminal according to any of
the above aspects and a network element according to any of
the above aspects is concerned.

35

15

Also, according to the present invention, a computer chip
comprising at least a signal processor according to any of
the preceding aspects is concerned.

5   According to the present invention, this object is for
example achieved by
a signal processing method for performing Fast Fourier
Transformation, FFT, of M_R, $M_R > 1$, input data streams
supplied in parallel, comprising the steps of multiplexing
10   the M_R input data streams to a multiplexed data stream,
performing Fast Fourier Transformation of the multiplexed
data stream and outputting the transformed data stream,
demultiplexing the transformed data stream to M_R
transformed output data streams, characterized by each of
15   the M_R input data streams contains a number of $N=2^k$
samples, performing FFT transformation using a pipeline of
k stages with a respective feedback path imposing a delay
on the samples per each stage of the pipeline and
controlling the performing of the FFT transformation by a
20   first and second internal control signals, and by imposing
a delay of $M_R*N/2^i$ samples on the samples in the feedback
path of an ith stage, $1<=i<=k$, of the pipeline, clocking
the first internal control signal M_R times faster compared
to a clock rate at which the samples of the M_R streams are
25   supplied, and clocking the second internal control signals
M_R times slower compared to the first internal control
signal.

According to advantageous further developments of the
30   signal processing method,
       - multiplexing is accomplished such that the M_R input
data streams are multiplexed per data sample of the input
data streams and demultiplexing is accomplished such that
the transformed data stream is demultiplexed per data
35   sample of the transformed data stream;

16

- clocking to the multiplexer and demultiplexer is performed at a rate M_R times the clock rate of the supplied streams;

5   - the Fast Fourier Transformation processing is based on a Radix-2 Single-path Delay Feedback algorithm;

- the pipeline of processing stages for the Fast Fourier Transformation is composed of Butterfly stages of types I and II (BF2I, BF2II);

- the first stage of the pipeline receiving the

10   multiplexed data stream is a Butterfly stage of type I for even and odd total numbers of k.

Still further, according to the present invention, a computer program product for a computer, comprising

15 software code portions for performing the steps of any one of the above method aspects when the program is run on the computer is concerned.

In this regard, the computer program product advantageously comprises a computer-readable medium on which the software

20 code portions are stored.

According to the present invention, at least the following advantages can be achieved compared to pre-existing concepts:

25

The present invention concentrates on the Fast-Fourier transformation in MIMO-OFDM systems. The proposed FFT-structure and method enables a transformation process of several incoming data streams in parallel.

30

However, the present invention is not limited to OFDM systems but can be applied to other scenarios in which parallel input data streams are to be subjected to FFT. For example, it can be applied for frequency domain filtering

35 at multiple antenna receiver or transmitter. For example,

as examples of OFDM systems, it can be applied to WLAN
systems or other communication systems such as those
currently investigated and referred to as 3.9G and 4G radio
communication systems.

·5

The new multi-stream FFT structure offers a reduction of
the computational complexity up to one FFT for all parallel
data streams. On the contrary to the above introduced
successive implementation, this strategy requires less

10   memory ($4N_{FFT} -4$ complex symbols) at same computational
complexity.

The proposed architecture combines the optimum properties
of parallel and straightforward successive multi-stream

15   FFT. The proposed architecture/method has the same
computational complexity as the straightforward successive
FFT solution. Thus, the gain is equal to the number of
parallel streams (M_R) compared to the parallel solution. It
has the same memory consumption as the parallel FFT

20   solution. The difference to the straightforward successive
solution is more than $2.5N_{FFT}$ complex symbol memory. The
lower complexity results in lower costs. It can be realized
with very little control "overhead" by merely adjusting
buffer capacity in the feedback paths and adjustment of

25   timing for the control signals.

The significant reduction of the number of FFT blocks
results in a corresponding reduction of cost for MIMO
systems. Thereby, about 1/3 of memory reduction compared to

30   a successive implementation using $R2^2SDF$ pipeline
architecture becomes possible by improved data processing
timing and feedback path delay adjustment.

18

The concept underlying the present invention can be applied
to all SDF pipeline FFT architectures with feedback delay
elements in the single delay feedback path.

5   Together with an increased processing rate of the FFT a
slight increase in power consumption is to be expected, if
the FFT is for example implemented in CMOS technology.
However, the particular hardware realization is not limited
to CMOS, but other technology concepts known for
10   implementing digital circuits are likewise applicable.

Brief description of the drawings

The present invention will be described with reference to
15   the accompanying drawings in which

Fig. 1 shows a signal flow graph of a Butterfly structure
with decomposed twiddle factors;

20   Fig. 2 shows a Radix 2^2 DIF FFT signal flow graph for N=16
samples;

Fig. 3 shows a Radix 2^2 SDF pipeline FFT architecture for
N=16 samples;

25
Fig. 4 shows an internal structure of a Butterfly stage of
first type, BF2I, with signals input thereto being divided
into real and imaginary part;

30   Fig. 5 shows an internal structure of a Butterfly stage of
second type, BF2II, with signals input thereto being
divided into real and imaginary part;

Fig. 6 shows a block circuit illustration of a parallel
35   symbol FFT transformation architecture;

Fig. 7 shows a block circuit illustration of a successive
symbol FFT transformation architecture;

5   Fig. 8 shows a timing diagram for the successive FFT
transformation architecture of Fig. 7. Note that this
diagram shows the timing for the first stage for the input
signal of the FFT length N only. However, the timing for
the following butterfly stages can be derived based on the
10   timing of the first stage. For this reason, according to
the stage i, the N value has to be adopted to $N=2^{k-(i-1)}$;

Fig. 9 shows a block circuit illustration of an embodiment
of a multi-stream FFT architecture, as applicable for
15   example to a 4 antenna MIMO receiver; and

Fig. 10 shows a basic timing diagram for the FFT
architecture according to the embodiment shown in Fig. 9.
Note that this diagram shows the timing for the first stage
20   for the input signal of the FFT length N only. However, the
timing for the following butterfly stages can be derived
based on the timing of the first stage. For this reason,
according to the stage i, the N value has to be adopted to
$N=2^{k-(i-1)}$;

25

Fig. 11A and 11B show details of the data control in terms
of control signals applied to a butterfly stage of type
BF2II according to prior art (Fig. 11A) and the present
invention (Fig. 11B), respectively;

30

Fig. 12 shows details of timing relations between the
control signals shown in Fig. 11A and applied according to
the prior art;

20

Fig. 13 shows details of timing relations between the
control signals shown in Fig. 11B and applied according to
the present invention;

5   Fig. 14A shows a block circuit diagram of a control module
according to the present invention, and
Fig. 14B shows a block circuit diagram of a modification of
a control module according to the present invention;

10   FIG. 15 shows parts of a system comprising at least one
terminal and at least one network element each of which
incorporates the FFT according to the present invention.

DETAILED DESCRIPTION OF THE PRESENT INVENTION

15

According to the present invention, basically, in N-by-M_R
MIMO systems, there are M_R data input streams in
parallel. (Note that this means here an N transmit and M_R
receive antenna system and N is not equal to the number N
20   of symbol samples to be subjected to FFT processing). For
this reason, an FFT architecture is also implemented which
is able to process several data streams simultaneously at a
rate M_R times the sample rate (of the individual data
stream). (This means, a clock signal clk' supplied to an
25   arrangement according to the present invention is M_R times
the clk signal applied to the prior art arrangement in
terms of frequency and 1/MR times in terms of period.)

Fig. 9 illustrates an FFT architecture for M_R=4 parallel
30   data streams and Fig. 10 shows the basic timing of the
signal processing, according to the present invention.

In the first step of the process, the M_R (M_R=4) data
streams $x_1(n)$, $x_2(n)$, $x_3(n)$ and $x_4(n)$ are multiplexed to a
35   single stream X(n) that is directly fed to the FFT pipeline

processor. For this reason, there is no need to introduce
any input buffer, which would have at least a size of M_R
times of the number N of data samples to be subjected to
FFT transformation. (N is also referred to as "FFT
5 length".)

For the transformation of the input x'(n), the known
architecture, according to the present invention, is
modified in respect of the subsequently outlined aspects.
10 Due to the four-fold amount of data (generally, M_R fold) at
each stage, the FIFO memory size in the feedback path of
each stage is extended by factor four (generally M_R). In
addition, since the same twiddle factors are used for each
of the four streams, the twiddle factors change four times
15 slower compared to the single stream FFT.

This means that the simple multipliers are maintained
active M_R times longer and also the factors W(n) are
applied M_R times longer.
20

Finally, the transformed data streams contained in an FFT
output stream X(k) are demultiplexed corresponding to the
multiplexing at the beginning of the FFT.

25 The overall memory size is $M_R(N_{FFT} -1)$. Comparing the before
described successive architecture, this approach requires a
significantly smaller memory size. Because of the
interleaved data processing within the FFT, there is no
need for buffering of the FFT inputs.
30

Table 2 shows the comparison of the successive multi-stream
FFTs. It can be seen that the new architecture reduces the
memory size of above $2.5N_{FFT}$ complex symbols at same
computational complexity.

35

22

Straightforward Successive Multi-stream FFT $M_R = 4$ $M_{FFT} = 1$	Successive Multi-stream FFT acc. to invention $M_R = 4$ $M_{FFT} = 1$
$6.5 N_{FFT} - 1$	$4 N_{FFT} - 4$

Table 2: Memory consumption of the successive alternative multi-stream FFTs.

Fig. 9 thus shows a signal processor for Fast Fourier
5 Transformation, FFT, of M_R, $M_R > 1$, input data streams
$x_i(n)$. In the example shown, $M_R = 4$, so that input data
streams $x_1(n)$, ..., $x_4(n)$ are supplied in parallel. The data
streams are fed to a multiplexing device MUX having M_R
(here $M_R = 4$) input terminals each receiving one of the M_R
10 input data streams x1(n), ..., x4(n). At an output terminal
x'(n) of the multiplexing device, the M_R input data streams
are output in a multiplexed manner. The multiplexed output
represents an interlaced (or interleaved) output of the M_R
data streams, i.e. data samples of M_R streams are
15 alternatingly output.

The thus obtained interlaced and/or multiplexed output data
stream x'(n) is fed to a Fast Fourier Transformation device
FFT. The FFT device is configured to perform Fast Fourier
20 Transformation of a data stream x'(n) supplied at an input
terminal thereof and to output the FFT transformed data
stream at an output terminal X(k) thereof. Thus, the input
terminal of the Fast Fourier Transformation device FFT is
connected to the output terminal x'(n) of the multiplexing
25 device MUX. The signal processor further comprises a
demultiplexing device DEMUX having an input terminal
connected to the output terminal X(k) of the Fast Fourier
Transformation device FFT. At M_R output terminals X1(k), ...,
X4(k) a respective one of M_R transformed output data
30 streams is output in a demultiplexed manner. (Note that
x(n) denotes the input signal in the non-FFT transformed

domain whereas X(k) denotes the resulting signal in the FFT transformed domain. In particular, k of X(k) is distinct from "k" used in connection with identifying the stages of an FFT applied.)

According to the present invention, such a FFT device is designed for each of the M_R input data streams containing a number of $N=2^k$ samples. Further, the Fast Fourier Transformation device FFT has a pipeline architecture composed of k stages with a respective feedback path including a single delay element per each stage of the pipeline architecture and is controlled by internal control signals clk', s, t, and w (not all individually shown in Fig. 9). The clock signal clk' is denoted as first control signal, and control signals s', t', w' are denoted as second control signals.

According to the present invention, the delay element in a feedback path of an ith stage, $1<=i<=k$, of the pipeline architecture imposes a delay of $M_R*N/2^i$ samples, first internal control signal clk' is clocked M_R times faster compared to a supply rate/clock rate of the supplied M_R sreams, and the second internal control signals s', t', w' are clocked M_R times slower compared to the clock rate clk' at which the FFT is operating..

In particular, the multiplexing device MUX is configured such that the M_R input data streams are multiplexed per data sample of the input data streams (interlaced) and the demultiplexing device (DEMUX) is configured such that the transformed input data stream is demultiplexed per data sample of the transformed data stream (de-interlaced).

A control signal (not shown) supplied to the multiplexer and demultiplexer is clocked at a rate of M_R*clk, which

means that it is operated at M_R times the clock rate clk / sample rate of the input data streams.

In a particular advantageous embodiment of the present
5   invention, the Fast Fourier Transformation device FFT has a Radix-2 Single-path Delay Feedback, R^2SDF, architecture. Also, the FFT device is clocked M_R times faster than the sample rate clk of an individual data stream of N samples. In connection with an $R2^2SDF$ FFT device, the pipeline
10   architecture of the Fast Fourier Transformation device is composed of Butterfly stages of types I and II (BF2I, BF2II).
In such a case, the first (input) stage of the pipeline architecture receiving the multiplexed data streams is a
15   Butterfly stage of type I for even and odd total numbers of stages. The internal structure and operation of BF2I and BF2II stages is as shown in Figs. 4 and 5, and only the timing of the control signals are different in connection with the present invention.
20

Fig. 11B shows details of control signals with a corresponding timing relation being illustrated in Fig. 13. Fig. 11B is substantially identical to Fig. 11A except that the control signals are denoted in addition with an
25   apostrophe to make clear that the control signals applied according to the present invention differ in the timing from those applied in the prior art arrangement.

Fig. 13 shows the timing relation there between.
30   In the lower part of Fig. 13, the signals z', w' and clk' are supplied. With each clock cycle clk', a new signal z' is supplied to the multiplier which is supplied with a corresponding weight (twiddle) factor w' which changes but after M_R cycles of clk'. In the upper part of Fig. 13 it is
35   shown that a sample x' of a repective one out of M_R

sequences of 1... N samples each (forming one OFDM symbol) is supplied with each clock cycle clk' in a multiplexed (interlaced) manner. Initially, the signal s' assumes a low level (s'=0) for the first $M_R*N/2$ samples. Thereafter,

5   starting with the interlacing of sample $M_R*N/2+1$, it assumes a high level until M_R*N samples of all streams of a symbol have been supplied. (Thereafter, a new OFDM symbol sequence starts with s'=0). As to the signal t', this signal assumes a high level for the first $M_R*3*N/4$ samples

10   and changes afterwards (starting with interlacing of samples $3*N/4+1$) for the last $M_R*N/4$ samples to the low level.

Thus, the second internal FFT control signals s', t', w'

15   are clocked M_R times slower compared to the clock rate clk' at which the FFT is operating, and the clock rate clk' at which the FFT is operating is MR times faster than the clock rate clk at which the samples of the MR streams are supplied. Speeding the clock rate clk' at which the FFT

20   device operates by a factor M_R adjusts the FFT clock rate to the number M_R of externally supplied data streams, and slowing the control signals s', t', w' down by a factor M_R compensates for this by adjusting the other internal control signals of the FFT to the new clock rate clk' at

25   which the FFT is operating.

As mentioned beforehand, it is to be noted that this diagram shows the timing for the first stage for the input signal of the FFT of length N only. However, the timing for

30   the following butterfly stages can be derived based on the timing of the first stage. For this reason, according to the stage i, the N value (based on which the timing is indicated) has to be adopted to $N=2^{k-(i-1)}$.

26

Fig. 14A shows a block circuit diagram of a control module
according to the present invention. As illustrated, a clock
rate clk of the M_R supplied streams is supplied to the
control module as well as an information on M_R as such.
5   Both of these can be fixedly configured to the FFT device,
or informed to the device during lifetime. In a first
frequency division block, the first internal control signal
of the FFT device clk' is generated by such that the first
internal control signal (clk') is clocked M_R times faster
10   compared to a clock rate (clk) at which the samples of the
M_R streams are supplied. This first internal control signal
is supplied to a control signal generation block of the FFT
device. Based on the supplied clock signal, second internal
control signals s, t, and w are generated, basically in the
15   manner as known from the prior art for controlling the
pipeline FFT architecture as described herein before, i.e
based on the number of clock cycles/samples of a single
stream processed. The first internal control signal clk' is
also passed to the pipeline architecture.
20
However, due to those (intermediate) second internal
control signals s, t, and w being generated based on clk',
the increased frequency thereof is to be compensated. This
is accomplished by a second frequency divider block. The
25   (intermediate) second internal control signals s, t, and w
are supplied thereto as well as the indication of M_R, and
an output of the second internal control signals s', t',
and w' is generated such that the second internal control
signals (s', t', w') are M_R times slower compared to the
30   first internal control signal (clk'). Then, also the
signals s', t', w' are supplied to the FFT pipeline
architecture.

Fig. 14B shows a block circuit diagram of a modification of
35   a control module according to the present invention. The

27

indication of M_R streams to be processed is supplied to a
FIFO Control block, where a memory control signal MEM_CTRL
is generated therefrom. The signal MEM_CTRL is then
supplied to the control section of e.g. a FIFO memory or
5   any other memory having FIFO capabilities within a feedback
path of a respective stage of the FFT pipeline structure.
As described above, according to the present invention, a
memory (e.g. FIFO) in a feedback path of the FFT pipeline
imposes a delay of $M_R*N/2^i$ samples on the samples in the
10   feedback path of an ith stage, $1<=i<=k$. This is based on
the assumption of a fixed number of M_R streams to be
processed which is known beforehand, i.e. at FFT device
production.

15   Fig. 14B now illustrates an example in which a FIFO or any
other memory is composed of a number of $j=1...M_{Rmax}$ memory
cells, each comprising $N/2^i$ memory locations for data
samples to be buffered. By virtue of the control signal
MEM_CTRL, a number of $M_R=x$ cells can be selected to be
20   actively used in the FIFO. Hence, data supplied at clock
rate clk' are output in a FIFO manner after $M_R=x$ memory
cells. This can be regarded as a FIFO than can be "tapped"
dependent on the control signal MEM_CTRL. Such feature
provides for increased flexibility of application of the
25   FFT structure in various environments, including SISO
($M_R=1$) as well as MIMO applications ($M_R=2...M_{Rmax}$). The
parameter M_R could be configured upon installation of the
FFT device, or could be transmitted in a special signal
(e.g. broadcast signal) and then detected at the FFT device
30   for self-configuration (or self-reconfiguration) of the
device. The only additional memory requirement would reside
in the feedback paths, but no buffers as discussed in
connection with the approach shown in Fig. 7 and 8 are
needed.

35

28

A signal processor according to any of the preceding
described aspects can advantageously form part of a network
element of a communication network. Still further, a signal
processor according to any of the preceding described

5   aspects can advantageously form part of a terminal
configured to communicate via a communication network.
Hence, the present invention also addresses a system
comprising at least one such a terminal and at least one
such network element, as shown in outline in Fig. 15. Fig.

10   15 shows an FFT according to the present invention being
implemented in a MIMO OFDM system comprising a Node_B as a
network element and a user equipment UE as a terminal. As
illustrated by the four ($M_R=4$) arrows, these communicate in
a MIMO scenario and in the illustrated example system, each

15   of them includes an FFT according to the present invention.
(Details of the FFT can be found in the respective other
figures of this application. Note that other components of
a terminal and a network element are not shown as they are
not essential for the present invention.)

20   Hereinbefore, the present invention has mainly been
described with reference to a hardware implementation as
e.g. usable in an ASIC (Application Specific Integrated
Circuit) or DSP (Digital Signal processor). The signal
processor can also be a signal processing device

25   implemented as a chip in semiconductor technology such as
CMOS, BiCMOS, or any other.

For a specific implementation of the invention, it is not
considered essential whether the invention is embodied as a

30   chip, as a signal processor device or as software code
portions as all these implementations are equally well
applicable and chosen according to the circumstances under
which the present invention is to be carried out. Thus,
whether a terminal or network element embodies the

35   invention as software code portion or as a chip or as a

29

signal processor device is not in the focus of the present application.

Nevertheless, the present invention may also be carried out
5 in terms of a signal processing method as software code portions running on a processor, or stored on a storage medium and thus adapted to carry out the method when run on a processor.

10 In this regard, its is to be understood that the present invention concerns a signal processing method for performing Fast Fourier Transformation, FFT, of M_R, $M_R > 1$, input data streams $(x1(n), ..., x_{M_R}(n))$ supplied in parallel, comprising the steps of multiplexing the M_R input
15 data streams $(x1(n), ..., x_{M_R}(n))$ to a multiplexed data stream, performing Fast Fourier Transformation of the multiplexed data stream and outputting the transformed data stream, demultiplexing the transformed data stream to M_R transformed output data streams, characterized by each of
20 the M_R input data streams contains a number of $N=2^k$ samples, performing FFT transformation using a pipeline of k stages with a respective feedback path imposing a delay on the samples per each stage of the pipeline and controlling the performing of the FFT transformation by a
25 first (clk') and second internal control signals (s', t', w'), and by imposing a delay of $M_R*N/2^i$ samples on the samples in the feedback path of an ith stage, $1<=i<=k$, of the pipeline, clocking the first internal control signal (clk') M_R times faster compared to a clock rate (clk) at
30 which the samples of the M_R streams are supplied, and clocking the second internal control signals (s', t', w') M_R times slower compared to the first internal control signal (clk').

30

Under the aspect of the method, multiplexing is
accomplished such that the M_R input data streams are
multiplexed per data sample of the input data streams and
demultiplexing is accomplished such that the transformed
5   data stream is demultiplexed per data sample of the
transformed data stream. Clocking to the multiplexer and
demultiplexer is performed at a rate of M_R*N, i.e. M_R times
the sample rate of an individual data stream. The Fast
Fourier Transformation processing is based on a Radix-2
10   Single-path Delay Feedback algorithm, wherein the pipeline
of processing stages for the Fast Fourier Transformation is
composed of Butterfly stages of types I and II (BF2I,
BF2II).

15   In this connection, the first of k stages of the pipeline
receiving the multiplexed data stream is a Butterfly stage
of type I for even and odd total numbers of k.

Accordingly, as has been described herein above, the
20   present invention proposes a signal processor for Fast
Fourier Transformation, FFT, of M_R, $M_R > 1$, input data
streams of 2^k samples each, supplied in parallel. After
multiplexing the input data streams in an interlaced
manner, the resulting stream is subjected to FFT. The FFT
25   device has a pipeline architecture composed of k stages
with a respective feedback path including a single delay
element per each stage of the pipeline architecture. The
delay element and timing signals are adapted to cope with
FFT processing of the multiplexed streams using the single
30   FFT device only. After processing, the FFT processed data
stream is demultiplexed.

Although the invention has been described in the context of
particular embodiments, various modifications are possible

without departing from the scope and spirit of the
invention as defined by the appended claims.

It should be appreciated that whilst embodiments of the
present invention have mainly been described in relation to
mobile communication devices such as mobile stations,
embodiments of the present invention may be applicable to
other types of communication devices that may access
communication networks. Furthermore, embodiments may be
applicable to other appropriate communication systems, even
if reference has mainly been made to mobile communication
systems.

List of abbreviations:

OFDM	Orthogonal Frequency Division Multiplex
SISO	Single Input Single Output
MIMO	Multiple Input Multiple Output
FFT	Fast Fourier Transformation
BF	Butterfly
CFA	Common Factor Algorithm
DIF	Decimation-In-Frequency
SFG	Signal Flow Graph
SDF	Single-Path Delay Feedback